



Sistema de Controle de Portão por conexão bluetooth

Eng. Da Computação - Curso: Sistemas Embarcados

Prof. Dr. Mário Guimarães Buratto

Aluno: Weverson Euzebio Forbes Silva. - Matricula: 684058

Relatório final

06/2023

1. Introdução

Este relatório descreve o projeto de desenvolvimento de um sistema de controle de servo motor por conexão Bluetooth, com o objetivo de controlar a abertura e o fechamento de um portão por meio de um aplicativo móvel. O projeto utiliza um microcontrolador ESP32 programado através da IDE Arduino, um driver TB6612FNG Duplo Ponte H de Motor DC ou Passo, um mini motor DC 1.5-3V, uma fonte ajustável para protoboard, uma protoboard e jumpers para protoboard.

2. Materiais Utilizados

Os seguintes materiais foram utilizados no projeto:

- Microcontrolador ESP32 programado através da IDE Arduino
- Driver TB6612FNG Duplo Ponte H de Motor DC ou Passo
- Mini Motor DC 1.5-3V
- Fonte Ajustável para Protoboard
- Protoboard
- Jumpers para protoboard

3. Procedimentos

A implementação do sistema de controle do servo motor por conexão Bluetooth para o portão envolveu os seguintes passos:

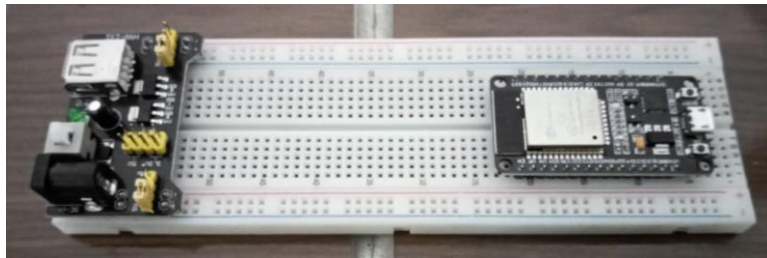
3.1. Conexão do Driver do Motor à ESP32 Primeiramente, foram identificados os pinos de controle do driver do motor (IN1, IN2, IN3 e IN4). Em seguida, utilizando jumpers, esses pinos foram conectados aos pinos GPIO da ESP32 na protoboard. Foi verificado o datasheet do driver para garantir a conexão correta dos pinos.



3.2. Conexão do Motor ao Driver Os terminais do mini motor DC foram conectados aos terminais de saída do driver, os quais foram rotulados como A1, A2, B1 e B2. Novamente, foi consultado o datasheet do driver para garantir a conexão adequada.



3.3. Conexão da Fonte de Alimentação A fonte ajustável foi conectada à protoboard para fornecer a tensão necessária para alimentar o driver e o motor. Foi importante verificar as especificações do motor e do driver para garantir o uso da tensão correta.



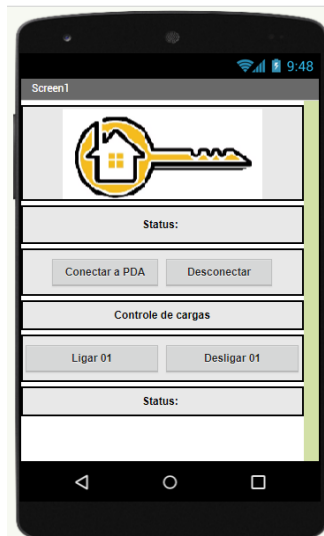
3.4. Programação do ESP32 Utilizando a IDE Arduino, foi desenvolvido o código para programar o microcontrolador ESP32. O código configurou a comunicação Bluetooth e interpretou os comandos recebidos do aplicativo móvel para controlar o servo motor. Foi utilizada a biblioteca "ESP32 Bluetooth Serial" para a comunicação Bluetooth.

```
testMotor2 | Arduino IDE 2.0.1
Arquivo  Editar  Rascunho  Ferramentas  Ajuda
ESP32 Dev Module

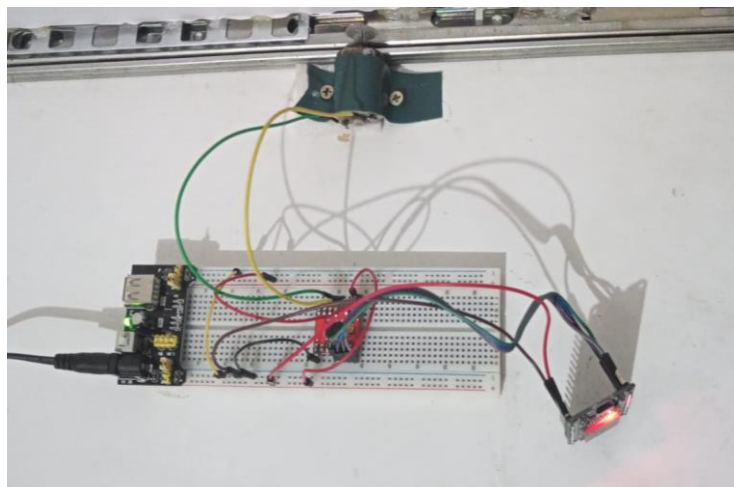
testMotor2.ino
1  #include <TB6612FNG.h>
2  #include <BluetoothSerial.h>
3
4  // 27 - Standby pin
5  // 14 - AIN1 pin
6  // 12 - AIN2 pin
7  // 13 - PWM pin
8  // To reverse forward motor direction, switch the AIN1 and AIN2 pin numbers.
9  TB6612FNG motor(14, 26, 27, 12);
10 // Tb6612fng motor(27, 12, 14, 13); // Reversed forward motor direction.
11
12 BluetoothSerial SerialBT;
13
14 void setup() {
15   Serial.begin(115200);
16   motor.begin();
17   SerialBT.begin("ESP32-MotorController");
18
19   pinMode(26, OUTPUT);
20   pinMode(27, OUTPUT);
21 }
22
23
24
25 /*void loop() {
26   if (1==1) {
27     char command = SerialBT.read();
28     switch (command) {
29       case 'F':
30         moveFrente();
31         break;
```

Link GitHub do algoritmo: <https://github.com/Weverson3000/Sistema-Controle-Portao-via-bluetooth-para-esp32.git>

3.5. Desenvolvimento do Aplicativo Móvel Utilizando o Android Studio (para Android), foi criado um aplicativo móvel que se conecta ao ESP32 por meio da conexão Bluetooth. O aplicativo foi projetado para enviar comandos de abertura e fechamento do portão para o ESP32.



3.6. Teste e Depuração após carregar o código no ESP32 e instalar o aplicativo móvel no dispositivo, foi realizado o teste do sistema. Verificou-se se o ESP32 estava corretamente emparelhado com o dispositivo móvel via Bluetooth. Foram enviados comandos de abertura e fechamento do portão pelo aplicativo móvel, e verificou-se se o servo motor respondia corretamente aos comandos.



4. Resultados e Discussão

O sistema de controle de servo motor por conexão Bluetooth para abertura e fechamento de portão foi implementado com sucesso. O aplicativo móvel conseguiu se conectar ao ESP32 via Bluetooth e enviar os comandos adequados para o motor. O servo motor respondeu corretamente aos comandos, resultando na abertura e fechamento do portão conforme desejado.

4.1. Problemas Superados e Soluções Adotadas

Durante a montagem do projeto, foram enfrentados alguns problemas que afetaram o desenvolvimento e o planejamento inicial. Esses problemas foram abordados e soluções foram encontradas para garantir o progresso do projeto. Os principais problemas e suas soluções são descritos a seguir:

4.2. Incompatibilidade de Tensão da Fonte de Alimentação

Durante um teste realizado, ocorreu a utilização acidental de uma fonte de alimentação de 24 volts, enquanto a fonte ajustável disponível suportava apenas 12 volts. Essa incompatibilidade levou ao mau funcionamento da fonte ajustável, o que impactou negativamente o planejamento inicial. Para contornar essa situação, foi necessária a aquisição de uma nova fonte ajustável com capacidade de fornecer a tensão adequada. Essa aquisição adicional resultou em um atraso no cronograma do projeto.

4.3. Problemas com a Biblioteca Utilizada para a Placa

Durante a implementação do código no microcontrolador ESP32, foi identificado que a biblioteca utilizada para a placa estava desatualizada, o que causou problemas no funcionamento correto do sistema. Para solucionar essa questão, foi necessário adaptar o algoritmo a partir de um exemplo disponível e remodelá-lo para atender às necessidades específicas do projeto. Essa abordagem permitiu contornar o problema e garantir o correto funcionamento do sistema.

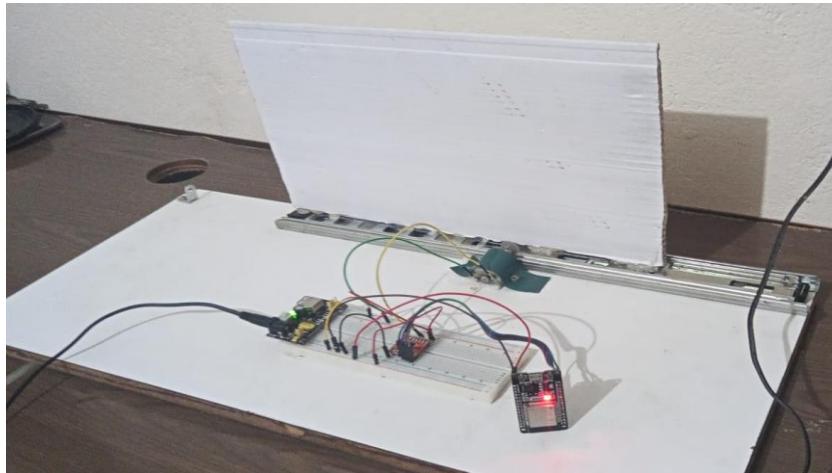
4.4. Dificuldades na Elaboração da Estrutura da Maquete

Ao montar a maquete para demonstração do sistema, surgiu uma dificuldade em relação à elaboração de uma engrenagem que fosse adaptável à maquete. Inicialmente, foi utilizada uma engrenagem de plástico, porém, essa escolha não se mostrou adequada, resultando em rotação incorreta do servo motor e impossibilitando o acionamento de abertura e fechamento do portão. Essa situação impactou inclusive na apresentação do projeto. No entanto, é importante ressaltar que o motor estava funcionando corretamente e conectado via Bluetooth ao aplicativo, realizando giros de acordo com as especificações.

5. Conclusão

Apesar dos problemas enfrentados durante a montagem do projeto, soluções adequadas foram encontradas para contornar as dificuldades e garantir o funcionamento adequado do sistema. A aquisição de uma nova fonte ajustável permitiu corrigir a incompatibilidade de tensão, enquanto a adaptação do algoritmo a partir de um exemplo e a remodelação do código solucionaram os problemas com a biblioteca desatualizada. Embora a dificuldade na elaboração da estrutura da maquete tenha afetado a demonstração do sistema, é importante destacar que o motor estava operacional e conectado corretamente ao aplicativo via Bluetooth. O projeto de desenvolvimento do sistema de controle de servo motor por conexão Bluetooth para abertura e fechamento de portão utilizando um microcontrolador ESP32 e um aplicativo móvel foi concluído com sucesso. Os passos envolvidos no projeto foram seguidos, desde a conexão dos componentes até a programação e o teste do sistema. O sistema demonstrou ser eficaz no controle do motor, permitindo a automação do portão de forma prática e conveniente por meio do aplicativo móvel. Esse projeto pode ser estendido adicionando um sistema de autenticação para garantir que apenas usuários autorizados possam controlar o portão e o alarme pelo aplicativo móvel. Também é possível integrar o sistema a uma rede para permitir o monitoramento remoto e o controle do portão e do

alarme. Além disso, o sistema pode ser conectado a uma câmera de segurança para visualizar imagens em tempo real e gravar imagens em caso de intrusão.



6. Recomendações

Com base no desenvolvimento do projeto, algumas recomendações podem ser feitas para melhorias futuras:

- Realizar testes de segurança e implementar mecanismos de proteção para evitar acidentes ou danos ao motor e ao portão.
- Aperfeiçoar a interface do aplicativo móvel, tornando-a mais intuitiva e amigável ao usuário.
- Explorar a possibilidade de adicionar recursos adicionais ao sistema, como detecção de obstáculos ou monitoramento remoto.

7. Referências

<https://www.usinainfo.com.br/blog/programar-esp32-com-a-ide-arduino-tutorial-completo/>

<https://www.usinainfo.com.br/driver-para-motor/driver-tb6612fng-duplo-ponte-h-de-motor-dc-ou-passo-4697.html>

https://www.technovationchallenge.org/wp-content/uploads/2015/01/tutorial-construcao_app-FaleComigo-pt.pdf

<https://embarcados.com.br/como-programar-o-esp32-na-arduino-ide/>